An Investigation into Self Organising Maps and Their Ability to Categorise Mixed Martial Artists and Their Fighting style

Reece Roberts

1601 7467

BSc Computer Science

21/04/2020

SCHOOL OF COMPUTING AND MATHEMATICS

Keele University
Keele
Staffordshire
ST5 5BG

Abstract

A self organising map is a machine learning technique used to classify and categorise information. Information is received as input data and then trained. Once trained, the input information is placed on a 2D plane comprised of nodes. The input data is allocated a node that has the most similarity to the data. The samples of input data were trained for a pre-determined set of iterations before a completed map is formed with every input sample stored inside it. During the training process, neighborhoods were set up on the map to contain all nodes that share similar values, the higher the similarity of two nodes, the smaller the distance on the plane they will be from each other. The Ultimate Fighting Championship is the mixed martial arts organisation in the world. Mixed martial arts were a combination of all fighting styles in one ruleset, but primarily comprised of two main fighting styles, striking and grappling. The Ultimate Fighting Championship owns a website that stores fighter statistics of every fighter past and present. These values were web scraped, filtered and tested in several self organising maps to test whether classification of strikers and grapplers could be derived from fighter's statistics. Tested with seven of the best strikers and seven of the best grapplers in the Ultimate Fighting Championship, the self organising map accurately clustered 85% of the fighters and had the ability to differentiate between a striking and grappling orientated martial artist.

Table of Contents

Table of Figures

1. Introduction

Artificial Intelligence (AI) and more specifically, Machine Learning, over recent years has become increasingly integrated within industries such as Sports Betting. Online gambling companies use AI to provide more accurate predictions and odds for their customers (Bet365, 2020). Machine learning algorithms intake a multi-variable set of input data and will use the data to train (either supervised or unsupervised), in order to produce an accurate output. As machine learning evolves and develops into a more precise and reliable algorithm for predicting, it can become a lucrative acquisition for the betting industry that is already worth £14.4 billion in the United Kingdom alone (Gamblingcommission.gov.uk, 2020).

AI can also provide a competitive edge within sports; athletes constantly study methods to gain a competitive edge in an industry where the smallest margins of error can lead to dire circumstances, particularly in full contact sports. AI can be used to create live training routines that were intended to maximise the efficiency of workouts, therefore demonstrating the potential large-scale implementation for either betting or competitive sports industries (Chu *et al.*, 2019).

1.1 Mixed Martial Arts

Mixed Martial Arts (MMA) is a full contact sport that implements a wide variety of fighting techniques such as striking, kicking and grappling, derived from traditional martial arts (Kung Fu, Taekwondo and Judo) and modern western martial arts (Boxing, Muay-Thai and Brazilian

Jiu-Jitsu) (Lystad, Gregory and Wilson, 2014). Fighters typically will combine these fighting techniques together to produce their own fighting style. (Hirose and Pih, 2009)

MMA bouts will typically be set in either a ring or a cage and fighters will compete against opponents of a similar weight and ability, the fighters were given a set number of rounds in which they must either knockout or submit their opponent. If the fighter is unable to finish the bout within the allocated time, the winner will be determined by the officiating judge's scorecards that take into consideration striking, damage, grappling and ring control. Fighters can essentially use all their body to attack their opponent, however, headbutting, eye gouging, groin attacks, biting and hair pulling were typically forbidden and may result in disqualification (UFC.com, 2020).

The culmination of different martial arts is not a recent discovery, Greek soldiers dating back to 648 BCE would fight against each other in training, the ruleset for these bouts is the fight would only stop when an opponent acknowledged defeat or was unconscious. This style of competition was known as Pankration and was a popular event in ancient Olympics (Encyclopedia Britannica, 2020).

MMA was popularised again in the 20th century, the resurgence originated in Brazil in the form of Vale Tudo, a full contact combat sport. The innovators within Vale Tudo were the Gracie family, a family of Brazilian Jiu-Jitsu practitioners who challenged anybody from any

background to compete with them (Alonso, 2013). At the time, nobody could overcome the challenge of defeating the Gracie family, thus sparking the idea of different martial artists competing against each other. In 1993, the Gracie's organised the first MMA tournament in Denver, Colorado called Ultimate Fighting Championship 1 (UFC 1) where Royce Gracie won with his Brazilian jiu-jitsu background and sparked the creation of the modern MMA competition, which today, has hundreds of organisations across the world.

1.1.1 Striking

The striking aspect of MMA is a key component and features the use of attacks when the competitors were standing in the bout. Striking consists of punches and kicks, as well as elbow and knee strikes. As MMA bouts start standing up, fighters with a background in striking orientated disciplines, such as boxing and Muay Thai, have an immediate advantage as they control the pace of the fight (Spanias *et al.,* 2019). Fighters proficient in striking will seek to end the fight via a knockout/technical knockout or decision (James *et al.*, 2016).

1.1.2 Grappling

The objective of grappling is to control an opponent, either in a standing clinch or when grounded. Grappling is position dominant and submission orientated, meaning the focus is to dominate the opponent into a position where a submission can be acquired (Ratamess, 2011). Backgrounds focused on grappling consist of wrestling, judo and Brazilian jiu-jitsu.

1.1.3 The Ultimate Fighting Championship

The Ultimate Fighting Championship (UFC) is the largest MMA organisation in the world, the UFC is worth around 4 billion dollars with an average pay-per-view attendance of 440,000 per event (Statista, 2020). Since starting in 1993, the UFC has remained at the top of the industry with the highest caliber of athletes on its roster. The UFC holds nearly 45 shows a year and although there were other elite organisations (Bellator, Rizin and Premier Fight League), none have had the level of athletes and viewership that the UFC has (R. Robbins and E. Zemanek, Jr., 2017).

1.2 Machine Learning

Machine learning is an extension of Artificial Intelligence which allows for systems to learn from experiences without the necessity of explicit programming, meaning, the program learns from itself to react to new data appropriately (Alpaydin, 2020). Machine learning enables a program to receive an input dataset of x values, where the input data is then processed by an algorithm, which results in the training of the algorithm for x iterations. The purpose of training is to give the program the ability to process values of the same format as the original input values (**Figure 1)** (Brownlee, 2020).

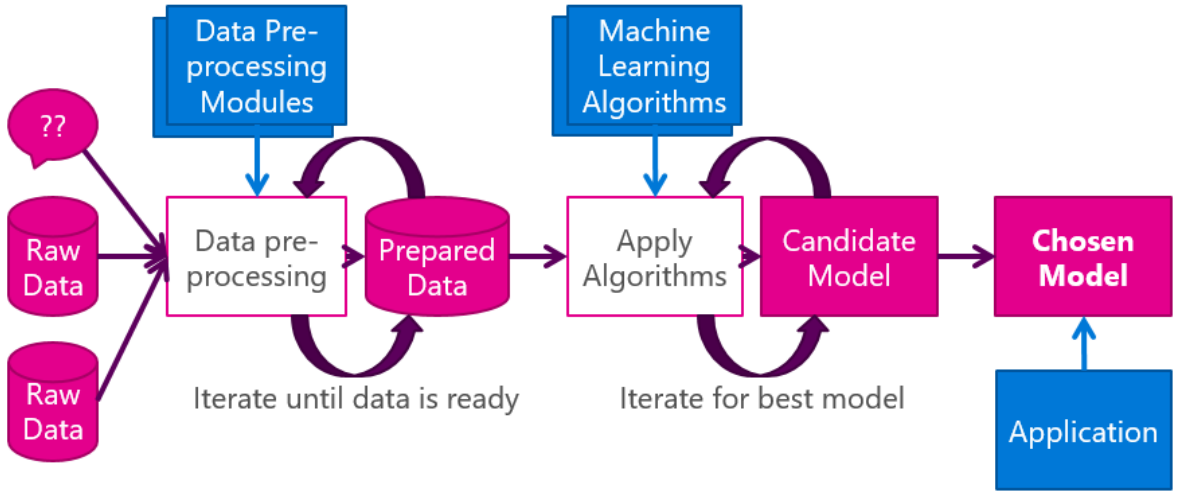

Figure 1. *A model of a Machine Learning Algorithm*. the data is prepared, fed forward and then the data's values were manipulated before selecting an output (Kearn, 2016).

Within the last decade there has been a high demand for the applications of machine learning in a diverse range of industry from finance to security. This is partly due to the accessibility

and advancements within the niche, meaning that new industries that previously wouldn't incorporate AI into their field now do with middleman hardware such as the internet of things (IOT) products (Ray, 2016).

An example of this is the agriculture industry; AI within agriculture has allowed for more efficient methods of produce, harvest and maintenance of crops. The AI can automate tasks, such as, monitoring fields and notifying farmers of any crop abnormalities on the farm. These advancements alleviate pressure for farmers and reduce overheads in the process (Gupta, 2019). The growth and diversity of what AI can do now compared to a decade ago explains why AI is expected to reach a global market size of 170 billion dollars by 2025 compared to 4 billion in 2016 (Patil, 2020). Machine Learning algorithms were typically divided between two types: unsupervised and supervised learning.

1.2.1 Unsupervised Learning

Unsupervised learning is the process of training data without any output information given to the algorithm, leaving the algorithm to figure the outcomes of the data independently. Unsupervised learning is commonly used to find patterns within data, which is particularly useful in scenarios where the aim is to cluster or classify data. An example of this type of implementation would be for loan applications, using previous data on loan applicants, a company can determine whether a new loan applicant is high risk or not (Springenberg, 2016).

Unsupervised learning is typically followed with a clustering algorithm of some form, allowing for the algorithm's output values to associate with one another to form relational patterns. Examples of these methods were K-Means clustering or Means-Shift clustering, k-means and means shift both search for the most optimal neighborhoods for the output data to be stored in, allowing them to only be associated with values that follow the patterns and attitude that the output value possesses (Sun, *et al.*, 2017).

1.2.2 Supervised Learning

Supervised learning an algorithm with a dataset of input and output values, whereby the output value is trained to have a direct relationship with the input values. The benefit of the supervised learning method is that once trained, the algorithm can give reliable output values when new input data is entered (Love, 2002).

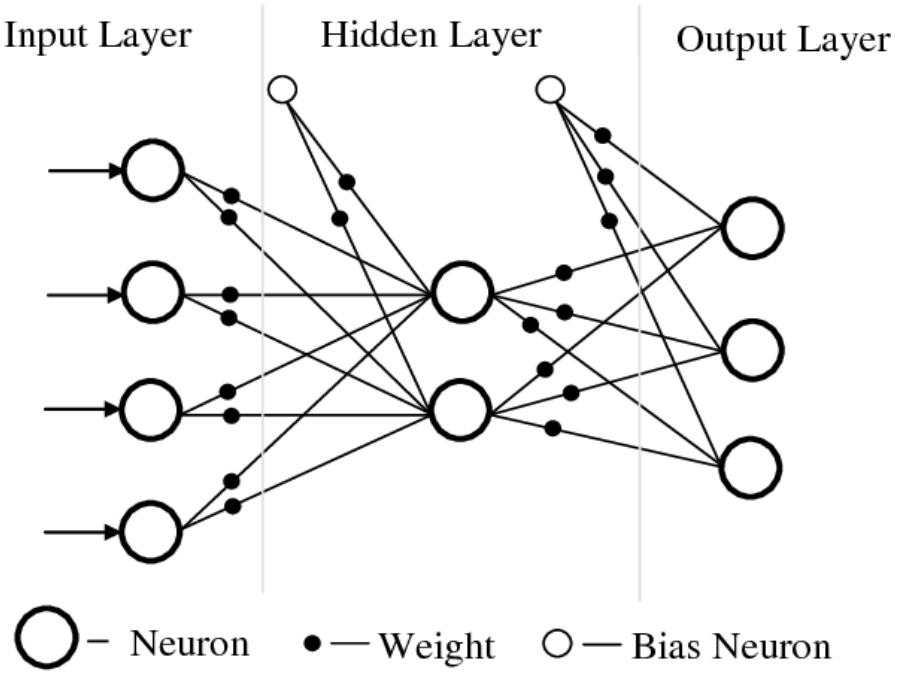

Figure 2**.** *A Basic Neural Network Architecture*. the input layer weights were passed forward to the hidden layer and the values were altered, before finally being fed forward to the output layer (Kulkarni, Londhe and Deo, 2017).

Supervised learning commonly follows the neural network model (**Figure 2)**. The model contains a column of input layer nodes with normalised values. The input values were then passed through to the next x columns, which were hidden layer nodes. Hidden layer nodes possess hidden weights, enabling the manipulation of the input values weights when the values were passed through the model. Once the input layer values have been fed forward to the hidden layer nodes, the values were finally passed forward through to the last column, the output layer. The output layer can comprise of one or many output nodes. The winning output node is the one that has most resemblance to the output value used for training. In practical applications, the output value will either represent a real value or a classification, such as ‘Dog’ or ‘Cat’.

1.3 Self Organising Maps

The self organising map (SOM) is an unsupervised learning model, the aim of the SOM is to plot values that were similar in attributes to one another on a 2D plane (Kohonen, 2013).

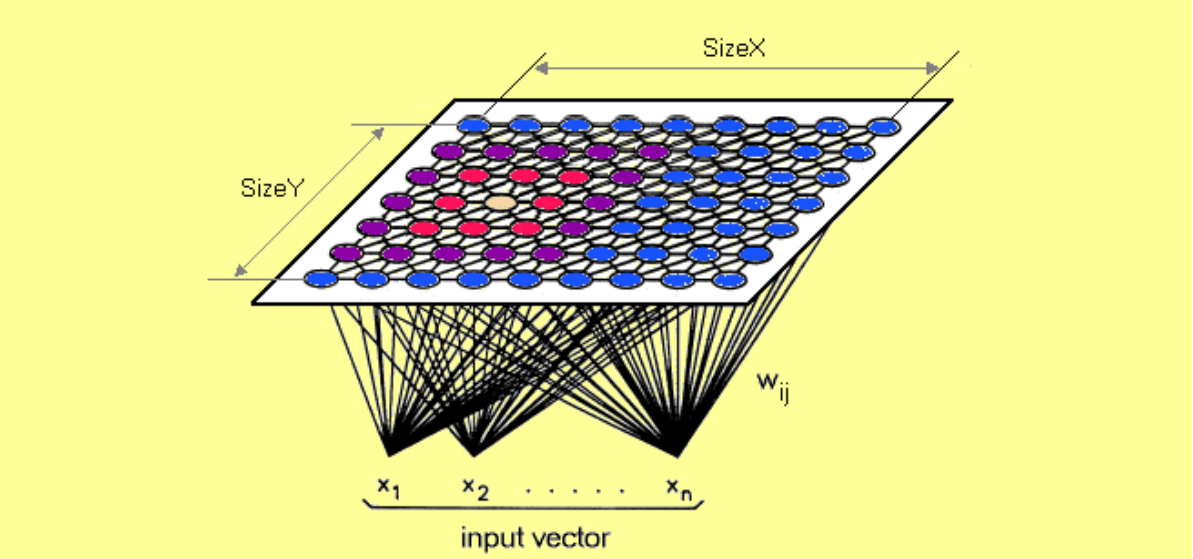

Figure 3**.** *A Self organising Map Architecture.* the input vectors values were assigned, the values on the plane were assigned. Then, the nodes on the plane layer were compared to find the node with the most similar values (Ralhan, 2018).

Originally established by Dr. Tuevo Kohonen, the SOM takes in an unknown input vector and places the vector into nodes that were normalised and assigned weights (**Figure 3)**. Once assigned the input vector's updated values will be compared to all nodes covering the 2D plane, whichever layer node has the closest value to the input vector wins and becomes the best matching unit (BMU) (Majumder, Behera and Subramanian, 2014). The neighborhood of the BMU is then calculated and determines what other nearby values on the plane were neighbors

(behave or possess attributes like one another). The final product is a map of values that should only have values that share the same characteristics within its neighborhood radius.

1.3.1 MiniSom

MiniSom is a Python package by Giuseppe Vettigli, MiniSom is a minimalistic and Numpy based implementation of the Self Organizing Maps (Vettigli, 2018). Using only a few lines of code to produce a SOM, MiniSom is versatile regarding unsupervised learning, as MiniSom can cluster and classify data along with other abilities, such as color quantization within an image. MiniSom is one of the most popular SOM packages out there currently. However, there were also other popular packages such as SOMPY or SimpSOM that use similar functions.

1.3.2 Neighborhood Functions

A neighborhood function is used within a SOM in order to determine what the neighborhood size and shape is. Neighborhood functions can be in the shape of any topological form, so long as it's winning node is in the centre. In the beginning, the neighborhood size will be the entire map. However, as the SOM is continuously iterated over, the neighborhood size reduces to only a couple of nodes after approximately 1,000 iterations (Meyer-Baese and Schmid, 2014). One of the most widely used neighborhood functions, particularly for SOMs, is the Gaussian function (Natita, Wiboonsak and Dusadee, 2016).

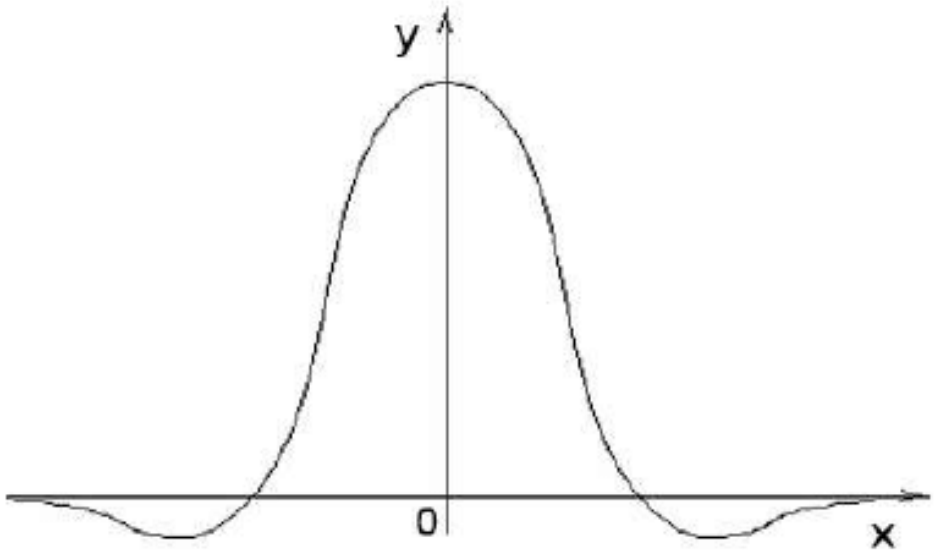

Figure 4**.** *Gaussian Neighborhood Function.* As values are further away from the centre point, the less the neighborhood function will manipulate the weights (Kajan, 2014)

The Gaussian neighborhood function adapts the weights of nodes that were closest to the winning node during the training process. Gaussian has less of an effect on nodes that were further away from the winning node, Gaussian specifically focuses its effect on nearby values. The benefit of using the Gaussian neighborhood method is that it adapts the neighborhood radius to allow for an even adaption rate of winning nodes in the local area (Kajan, 2014)

.

1.3.4 Uses of SOMs

SOMs can be effective in practice, the approach of organising data unsupervised means it can understand patterns within the data that could be otherwise ignored. Classification and cluster problems were best suited for SOMs. Security procedures such as fraud detection can be

identified with SOM architecture. With historical data available a SOM can successfully recognise fraudulent patterns within a system and flag it up (Olszewski, 2014). SOM architecture can also be implemented within a sport setting. Within football, a SOM can be trained in order to interpret whether a play will be long or short within the match (Grunz, Memmert and Perl, 2012).

1.4 Web Scraping

Web scraping is the method of gathering specific information from a website without having to do it manually. The principle behind web scraping is to extract data from a web page, but only getting the essential data (Vargiu and Urru, 2012). The benefit of this method is that data manipulation can be automated and done on multiple web pages, which in turn, will reduce the time spent attempting to extract webpage data.

Beautiful Soup is a python package that cleans up the HTML file of a web page and presents the page as a nested data structure (BeautifulSoup, 2020), resulting in clearer and more accessible code for people to use to find and scrape data from a web page. Beautiful soup changes HTML files into a Document Object Model (DOM) tree and then will call its search and editing functions to complete the task (Zheng, He and Peng, 2015).

1.5 Cluster Analysis

Cluster analysis methodically classifies objects that were like one another in character or attribute (Chu, 1989). Cluster analysis is naturally a part of the SOM process, as neighborhoods

themselves were clusters of information comprising of values that have an affinity towards one another and clustering itself is inherently unsupervised. A popular clustering method used for SOMs is k-means clustering. K-means is used as a method to automatically partition a data set into k cluster centres (Gan and Ng, 2017), the centres were then refined iteratively until all values on the plane were within a k-cluster. K-means can become an issue when the desired number of clusters is unknown, as it can then lead to sub clusters rather than a single cluster which may not be desirable when attempting broad classification.

1.5.1 Orange Data Mining

Orange Data Mining is an open-source software that implements a variety of machine learning and data visualisation tools. Orange is used primarily for data mining but is flexible in use. Orange's interface possesses tools for supervised and unsupervised learning models that were accessible without prior programming knowledge, this makes Orange a powerful tool for novices, but also simplifies a lot of processes for experts.

1.6 Aims

The main aims for the investigation were:

1. produce a dataset of UFC fighters scraped from the UFC's statistics website.
2. Test whether a self-organizing map, comprising of UFC athletes can have the ability.
3. To be able to visualise the self organising maps.
4. Use clustering algorithms to visibly display where the clusters are, and where elite strikers and grapplers lie in the plots in order to determine if the SOM architecture can differentiate between striking based and grappling based fighters.

2. Methods and Materials

2.1 Web Scrape

Fighter data was taken from ufcstats.com which stores fight and fighter analytics from every fight in UFC history. The fighters page, in alphabetical order lists all the UFC fighters, both past and present. Situated at the top of the page were links from A-Z to filter the fighters to only display fighters with the same first character in their surname. Each fighter page contains their physical attributes, previous bouts and their career statistics, the statistics were broken down into eight categories **(Table 1)**.

Table 1. Table displaying the abbreviation and description of UFC fighter's career statistics.

Career Statistic	**Description**
SLpM	Significant strikes landed per minute
Str. Acc	Significant striking accuracy
SApM	Significant strikes absorbed per minute
Str. Def	Significant strike defence (% of opponents strike that failed to land)
TD Avg	Average takedowns landed per 15 minutes
TD Acc	Takedown accuracy
Sub. Avg	Averages submissions attempted per 15 minutes

Each fighter has a unique ID number that is placed at the end of their page URL, the ID numbers do not have a uniform order to them nor a format, simply the ID is a mixture of random numbers of letters such as f77c68bb4be8516d (Yoel Romero) or 3c660c90d48fc80d (Tim Kennedy). With no order to the unique ids, an iterative loop cannot be created on its own to extract fighter

career statistics from each fighter's page. The first objective is to obtain every fighter's unique id.

Using Python in Atom script editor, an array was created to contain a list all the fighter's unique ids. The webpage listing the fighters contains links to each fighter's page and within the link is the fighter's unique id. The fighter list page doesn't possess an option to display all fighters on one page, it only has an option to display a list of all fighters sharing the same first character of the surname that the user selects. Therefore, a loop can be created to extract the unique IDs of the fighters with every first character of their surname (**Figure 5)**

```python
import requests
import pandas as pd
import re
from bs4 import BeautifulSoup

fighterURL = []

for i in (list(map(chr, range(97, 123)))):
    url = "http://ufcstats.com/statistics/fighters?char=" + \
        str(i) + "&page=all"

    response = requests.get(url)
    texth = response.text

    soup = BeautifulSoup(texth, 'html.parser')

    for link in soup.find_all('a', href=re.compile("http://ufcstats.com/fighter-details/")):
        text = str(link.get('href'))
        splitter = text.split("http://ufcstats.com/fighter-details/")
        Parsed = splitter[1]
        fighterURL.append(Parsed)
        print(Parsed)

    fighterURL = list(dict.fromkeys(fighterURL))


with open('UniqueIDS.txt', 'w') as filehandle:
    for listitem in fighterURL:
        filehandle.write('%s\n' % listitem)
```

Figure 5**.** *Web Scraping Loop for Fighter ID.* The Python code used to web scrape the unique IDs of every fighter within the database. The links to the fighter's pages were parsed in order to only store the ID at the end of them.

Data from every page that lists fighters from the hexadecimal range of 93 to 123 (A to Z in the hexadecimal system) was scraped and every instance of a unique ID in the web page was stored into the array. Once all the unique IDs were in the array, instances of duplicate IDs were removed, and the appended list was saved as a text file.

Once a file has been created for each fighters id, each fighter detail page can be scraped to retrieve the values of the career statistics. A request was made to retrieve the html data of every page using the URL path to the fighter details page with a unique ID appended at the end. Career statistics were then scraped, and the career values were formatted into real values. The number of fights each fighter has had in the UFC was also scraped. To retrieve this value, BeautifulSoup was used to find the number of instances a fighter's name had been included in a bout, not including the instance of their name in the page title. This number is then calculated and stored along with the career statistics into a data frame and exported as a csv file.

The data frame contained career statistics of 3,405 fighters, the data frame was filtered to remove any fighter with less than four fights recorded in the UFC and removed any fighter that had null values for every statistic. Once filtered, the data frame contained 1,228 fighters with each fighter having on average, ten UFC bouts.

2.2 Self Organising Map

In order to use the fighter's values within the self organising map, the fighter data was read into the program, normalised and converted into a NumPy array ready to be included in the som. To create the self organising map, MiniSom, a python package, was imported and a SOM object was created.

In order to validate the authenticity of the package, the iris dataset in the MiniSom documentation, which comprises of three different flower groups, was tested to see if the output values were the same as the ones in the clustering example of the documentation. MiniSom

documentation displays an example of clustering using the package alongside the iris dataset. When testing the clustering method in MiniSom, the results retrieved were the same as the ones referenced in the MiniSom documentation **(Figure 6).**

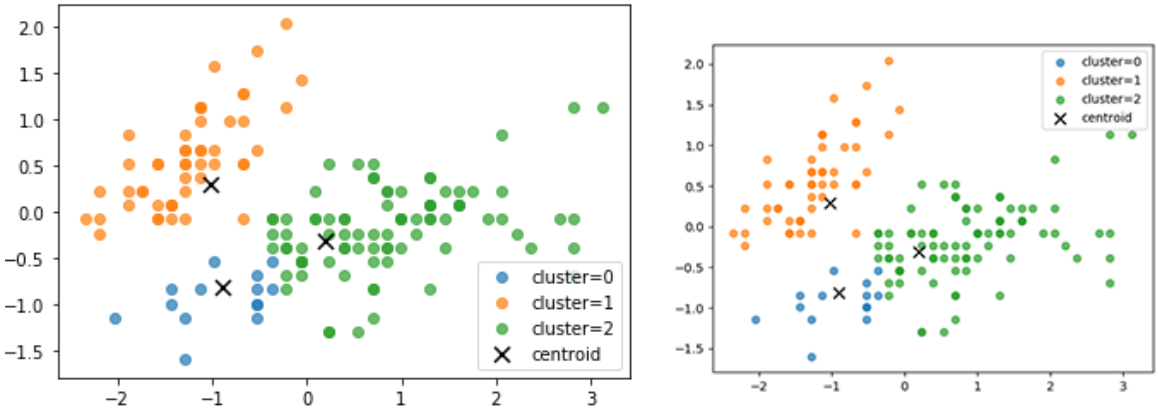

Figure 6**.** *Comparison between Documentation Output and Tested Output.* The SOM for the maps contained each fighter's number of fights included in the input vector of the som.

Once set up, the following parameters were initially set **(Table 2).**

Table 2. Initial values used to create the self organising map

Parameter	**Value**
Map Size	13x13
Sigma function	3
Learning rate	0.5
Neighborhood function	Gaussian
Random seed	10
Iterations	1000

The SOM was trained for 1,000 iterations, with the winning coordinates for each fighter stored and appended to the fighter data frame. Afterwards, a graph was produced using the Python package Matplot. The trained SOM's plane is then visualised on a graph to show the weight distancing on the SOM plane. Alongside the graph was a colour bar for reference.

The SOM was tested at different map sizes, iterations and number of input values used. The neighborhoods were tested at sizes 9x9,13x13 and 16x16, at iterations of 500, 1000, and 1500. The input values were tested without the inclusion of the number of fights each fighter had. The input values were also tested with only the percentage value inputs, and lastly, tested without the percentage value inputs.

2.3 Cluster Analysis

The fighter data frame equipped with each fighter's winning coordinates for every test is then input separately into Orange's data mining application. The fighters coordinates then were put through a k-means clustering algorithm and then finally visualised on a scatter plot. When using k-means clustering, the Orange application gives a silhouette score, which scores different cluster sizes between -1 or +1, the higher the score the closer matched objects were to their cluster. The cluster number with the highest silhouette score is used. The scatter plot is then saved and analysed to monitor which cluster seven renown strikers and grapplers were allocated to. The specialised fighters were chosen from an online website that has users rank the beast strikers and grapplers in mixed martial arts.

2.2 Packages

The following packages were used in the coding of the programs that were used in the investigation (**Table 3)**.

(Table 3). An overview of the python packages used in the programming for the investigation.

Package	**Description**
MiniSom	A minimalistic implementation of self organising maps.
NumPy	A library with support for multi-dimensional arrays and matrices, with functions to operate the arrays.
Matplot	Matplot is a library for creating static or interactive python visualisations.
Pandas	An open source tool for data manipulation and analysis.
BeautifulSoup/requests	Web scraping tools used to extract HTML and XML files from a webpage.
Sklearn	A machine learning library for Python. The library features classification and clustering algorithims.

3. Results

3.1 Including Number of Fights

The SOM was created and trained using the fighter's eight career statistics, as well as the number of fights the fighter has had in the UFC **(Figure 7)**.

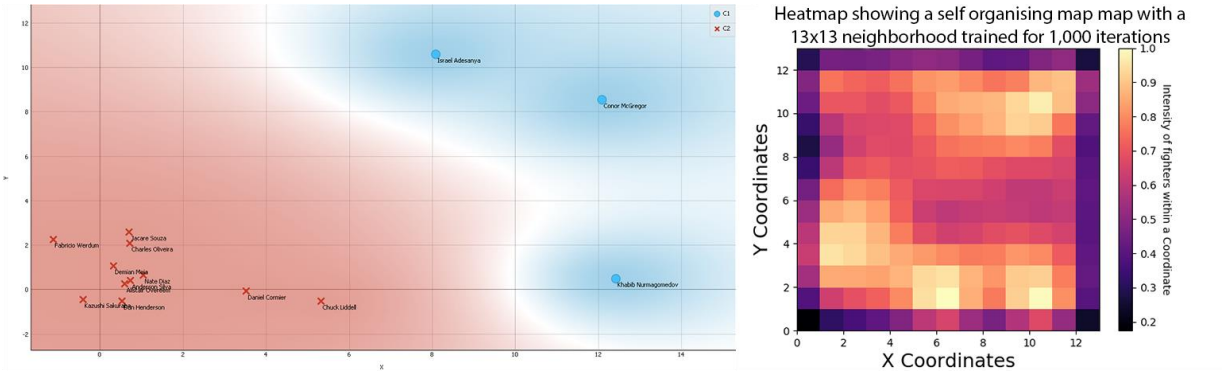

Figure 7**.** *The heatmap and cluster map for a trained SOM*. The SOM for the maps contained each fighter's number of fights included in the input vector of the som.

The cluster map labels where each fighter in the list of seven strikers and grapplers are. The fighters were skewed close together around the (1, 1) coordinate, nine of the fourteen selected fighters were around the coordinate. The k-means clustering selected two clusters for this SOM with a silhouette score of 0.714. The three fighters that were in cluster one had between eight and thirteen bouts in the UFC. The fighters in cluster two have between twenty-two and forty-seven bouts in the UFC.

The heatmap shows four areas where the distance between each weight shows 0.9 or above, this shows that the neighborhood connections were weak around this area (**Figure 7**). Around the perimeter of the map the distance between the weights is less at 0.4 or below, signaling strong character matches between the values in this region.

3.2 9x9 Neighborhood

The SOM parameters were the eight career statistics of the fighters, a 9x9 neighborhood and a training iteration number of 1,000 **(Figure 8)**.

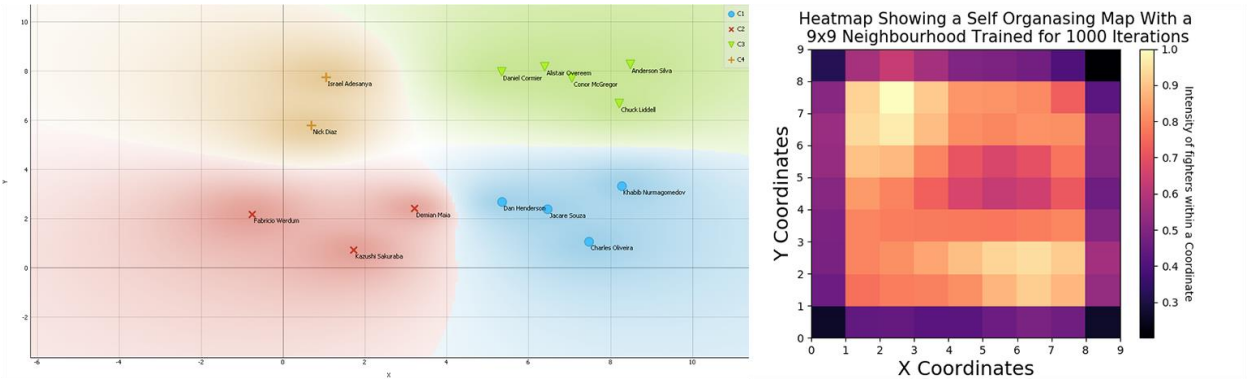

Figure 8**.** *The heatmap and cluster map for a trained SOM. with a neighborhood size of 9x9.* The SOM was trained for 1,000 iterations. The number of clusters was five with a silhouette score of 0.555.

The cluster map had a total of four clusters with a silhouette score of 0.555 retrieved from the k-means model. The strikers were clustered within clusters one and four, one striker (Dan Henderson) was an outlier plotted in cluster 1. The grapplers were clustered in clusters two and three, with one grappler (Daniel Cormier) plotted in cluster 2. The heatmap has two areas with distance scores above 0.9, around the corners of the map the

weight distance is 0.4 or below.

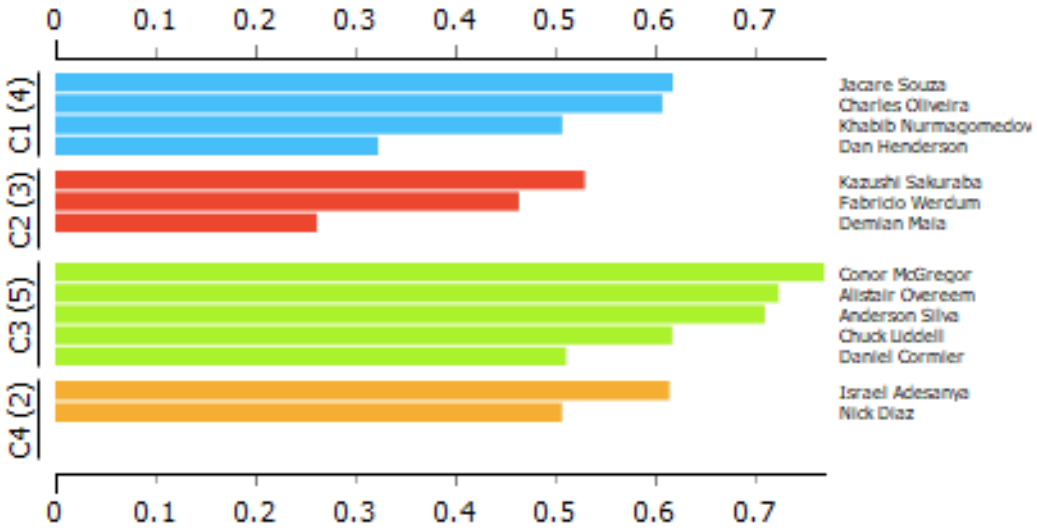

Figure 9**.** *The silhouette scores for each of the four clusters in the scatter plot.* The strikers in cluster three had the most similarity between each other, compared to the other clusters.

Cluster three had the strongest score out of the four clusters, four of the five fighters in the cluster had scores higher than the silhouette score for k-means cluster, the one with the lower score is the outlying grappler (Daniel Cormier). This indicates that the strikers were closely matched with one another in attributes (**Figure 9)**. Four of the seven strikers were clustered together, and three of the seven strikers were clustered together (two clusters both have three grapplers per cluster).

3.3 13x13 Neighborhood

The SOMs parameters were the eight career statistics of the fighters, a 13x13 neighborhood and a training iteration number of 1,000 **(Figure 10)**.

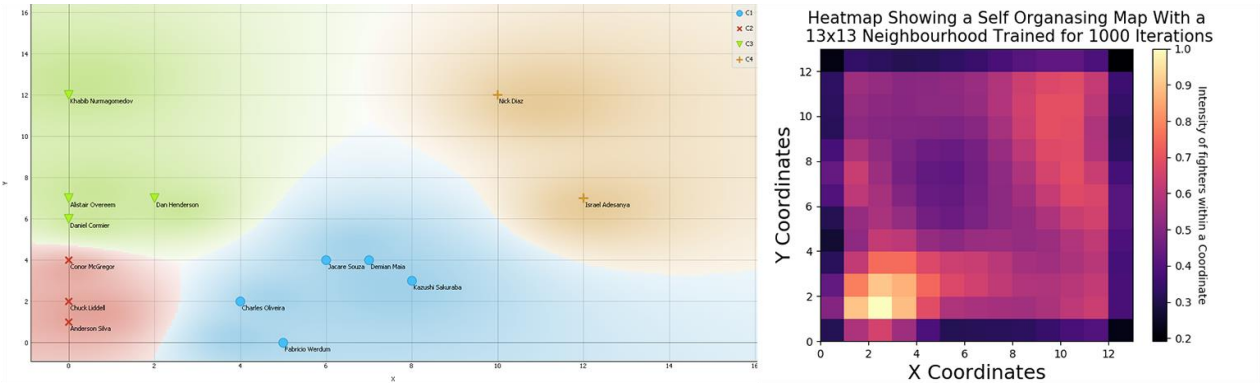

Figure 10**.** *The heatmap and cluster map for a trained SOM with a neighborhood size of 13x13.* This SOM was trained for iterations. The number of clusters was four with a silhouette score of 0.467.

The cluster map had a total of four clusters with a silhouette score of 0.467 retrieved from the k-means model. The strikers were clustered within clusters two, three and four. The fighters in cluster three were plotted close to the strikers in cluster two. The two strikers in cluster four were further away from the other fighters plotted in the top right of the map. The heatmap shows one area in the map with a weight distance of 0.9 or above, none of the fighters were affected by this area. The perimeter and centre of the map have low weight distances of 0.4.

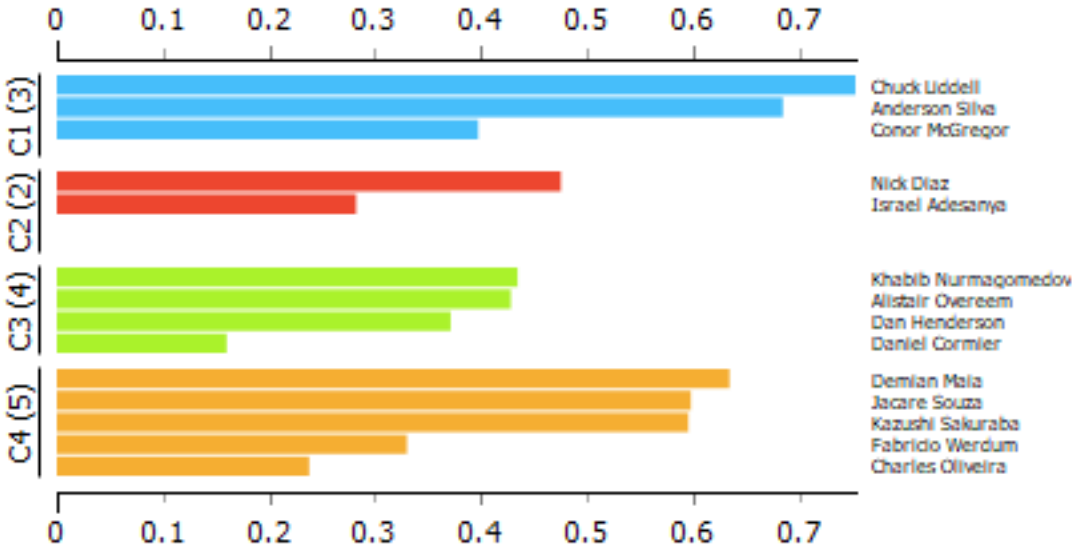

Figure 11**.** *The silhouette score of the selected strikers and grapplers*. The strikers in cluster one had the most similarity between each other. However, cluster four had more grapplers, three of the five had relatively high similarity between each other.

The grapplers clustered together in larger numbers, with five of the seven grapplers in cluster four. The other two grapplers were situated in cluster four. The strikers clustered mainly in cluster one, with three of the seven strikers plotted here, two fighters were in cluster two and the last two were in cluster three (**Figure 11)**.

3.4 16x16 Neighborhood

The SOMs parameters were the eight career statistics of the fighters, a 16x16 neighborhood and a training iteration number of 1,000 **(Figure 12).**

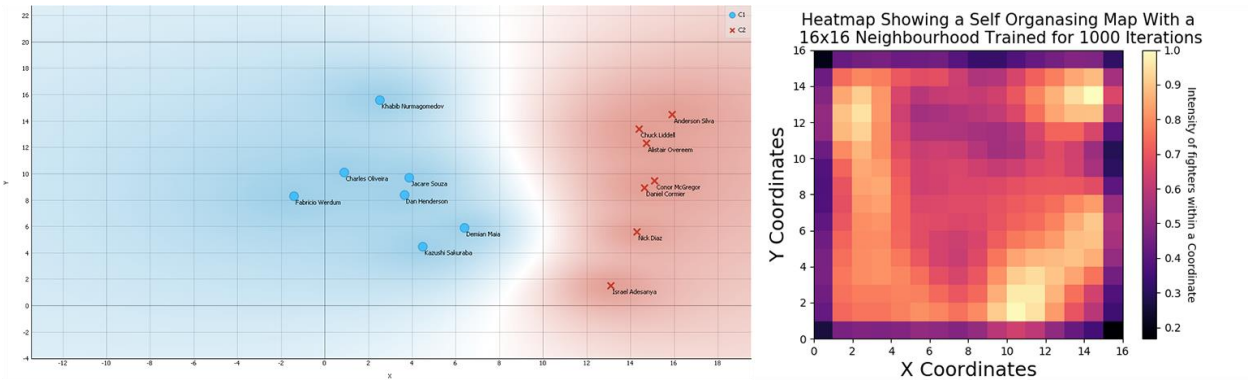

Figure 12**.** *The heatmap and cluster map for a trained SOM with a neighborhood size of 16x16.* The SOM was trained for 1,000 iterations. The number of clusters was two with the highest silhouette score of 0.577.

The k-means method determined the strongest cluster was a cluster size of two, for both the strikers and the grapplers. The fighter's clusters had six of their seven fighters in the neighborhood, so both had one outlier, for strikers it was Dan Henderson and for the grapplers it was Daniel Cormier. The heatmap displays space within the SOM with large areas of the map possessing high weight distances.

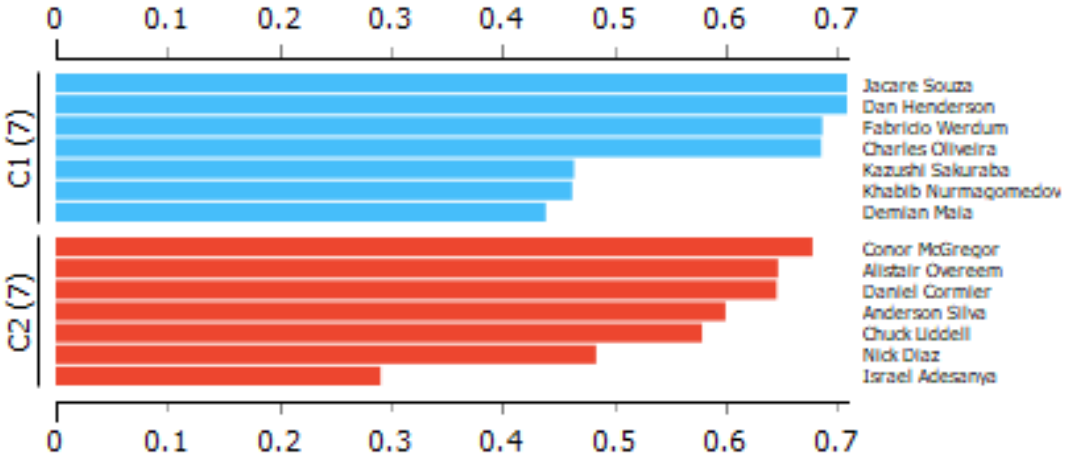

Figure 13**.** *Silhouette scores for the fighters in the som.* The silhouette score for the SOM was 0.577, nine of the fourteen fighters had higher scores than this.

The silhouette scores show that there is a high affinity between these fighters that were clustered together, both outliers for the strikers and grapplers have high silhouette scores in relation to the cluster that they are, indicating strong character similarity with the opposing fighting style (**Figure 13)**.

4. Discussion

4.1 Prerequisites

When the fighters were initially scraped from the web page, the original dataset would've introduced skewed and inaccurate data for the SOM. Some of the fighters who were in the UFC database possessed career statistics that were null values, hence there was no way of distinguishing whether the fighter was a striker or a grappler with no career data. Ultimately, these fighters had to be removed from the dataset in order to improve the reliability of the results.

Mixed martial bouts, like all sports, can be unpredictable; fighters deemed underdogs by the bookmakers can execute surprising victories, can fight in a different style depending on who their opponents are, and can display poor performances that don't accurately represent the fighter's quality. These scenarios, without much history on a fighter, can lead to inaccurate clusters and classifications. For example, UFC fighter Donald Cerrone fought against Conor McGregor in a Welterweight bout. Donald Cerrone, who has an above average striking output (4.33 significant strikes landed per minute, compared to the fighter dataset average of 3.45) didn't land a single punch for the entirety of the fight.

From the original dataset, the average number of fights a fighter had in the UFC was 4.49. In order to give a fairer overview of what each fighters default fighter is, any fighter with less

than four fights was to be stripped from the dataset. In doing this, the remaining 1,227 fighters would have reasonable experience within the UFC and had the time to time to in some degree display what their primary fighting style is.

The central neighborhood used for the investigation had a size of 13x13 nodes. The MiniSom documentation states, it is ideal to set the size of the neighborhood based on the size of the input data samples, the map should contain 5*sqrt(N) nodes where N is the number of samples from the input data (Vettigli, 2018). For the fighter dataset, the number of samples equated to 1,227, therefore, the number of nodes for the grid was 175, ensuring the 13x13 size was adequate. To monitor the effects of the SOM size on the final output, 9x9 and 16x16 node SOM's were tested as well, in order to see if they could have a higher or lower silhouette score than the recommended size.

The MiniSom package hosts examples of different uses of the SOM including clustering, classification and outlier's detection. As the investigation was to see whether a SOM could group together data with similar attributes, the parameters used in the example SOM to classify the example's iris dataset, were the same as what was used in the SOM's used in the investigation (Vettigli, 2018).

For cluster analysis, k-means clustering was selected as the method to cluster the fighter's dataset, this was due to its flexibility to converge to any number of clusters that were designated

to it (Ultsch, Morchen, 2005). The k-means model, along with the silhouette score, would show in k clusters how much the trained values match with the cluster they reside in.

The SOM's were tested at different iterations to see the effects of changing the number of iterations when training the SOM. When training, the neighborhoods should already be refined by 1,000 iterations (Meyer-Baese and Schmid, 2014). To see whether this iteration size lead top optimal performance, iterations of 500 and 1500 were also tested to see the affect it had on cluster quality.

Fighting disciplines vastly range in techniques and combat fundamentals, nonetheless, when in a combat situation the two fundamental modes of attack were striking and grappling (Spanias *et al*., 2019). The difference between striking and grappling methods were the techniques and principles that were exclusive to them. The two fighting styles comprise of all traditional and western martial arts. To begin to classify what fighters' styles were in MMA, first the differentiation between strikers and grapplers must have been completed (James *et al.*, 2016).

For the evaluation of the results, fighters were selected to represent the striking and grappling based fighters. There is no ranking provided by the UFC, so to gather a selection of elite fighters from each discipline, a community centred ranking list was used to gather the fighters. Tapology.com collects rankings from users to create a consensus rankings list for both the best

strikers and grapplers in UFC history. Fighters were selected from the top 20 of each list to produce a testing dataset.

4.2 Results of Testing

The tests showed that the SOM that distinguished the difference between strikers and grapplers the most, was the SOM with a map size of 16x16 nodes trained at 1,000 iterations. The SOM clustered together six of the seven strikers and six of the seven grapplers, possessing an accuracy of 85.7%. The silhouette score of 0.577 was the second highest score of all the results.

For this investigation, following MiniSom documentation, the optimal neighborhood size is 13x13 nodes. However, the 16x16 SOM recorded a 0.12 higher silhouette score than the 13x13 SOM, as well as partially successfully keeping the strikers and grapplers in separate clusters. A factor for this is the 16x16 SOM optimised for two clusters, which is ideal when attempting to differentiate between two styles. Whereas the 13x13 model was optimised to four clusters, making the classify the clusters much more complicated. The 16x16 SOM's increase in space may allow for the k-means clustering to reduce the neighborhood size more accurately with regards to classifying fighters (Lamb and Croft, 2016).

When initially testing the input vectors, the number of fights value was removed from the input vector due to high sensitivity when SOM training. The difference of spread on the charts between the same test with and without the number of fights value was significant. Before, nine

of the fourteen fighters were within a one node radius from another. After the number of fights were removed, the values spread out on the map much more evenly (**Figure 14)**.

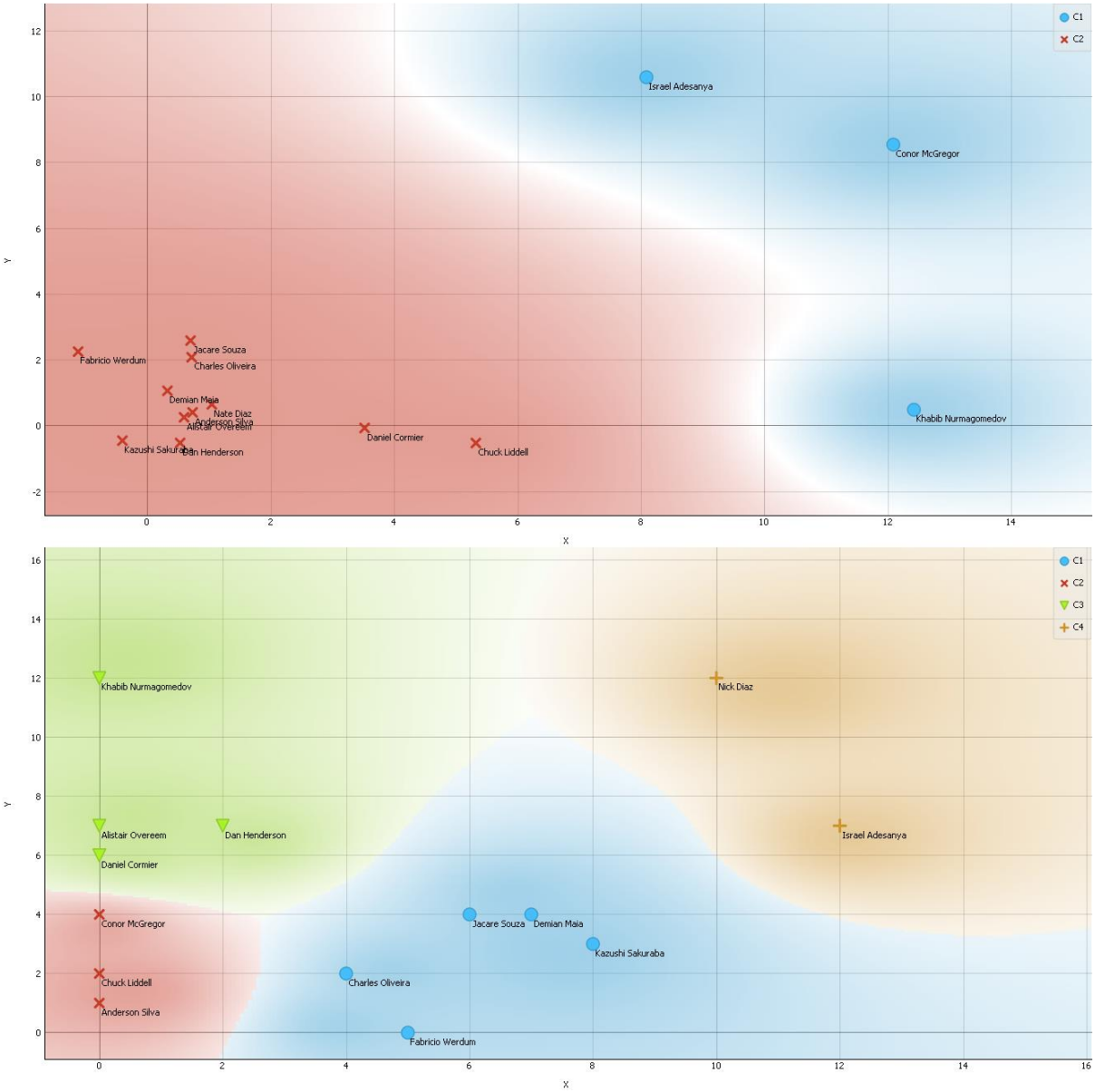

Figure 14**.** *A comparison between using and not using the number of fights value from the fighter dataset.* The top graph is the SOM when number of fights were included, compared to the bottom graph, which omitted the value. The graphs visually display the way fighters were plotted based on their number of fights and how fighters with a high number of UFC bouts were clustered together.

The number of fights value was ultimately deemed unnecessary for the classification of a striker or grappler, since fighters from both backgrounds can have any number of fights in the UFC, their fighting style does not inhibit this.

Within the tests were a couple of outliers that would repeatedly fall into the striker's clusters as opposed to the grappling clusters. Daniel Cormier and Dan Henderson both showed more similarity with their opposing style than their own. Daniel Cormier is classified as a grappler, presumably due to his elite wrestling background, placing third in the freestyle wrestling world championships (Sesker, 2014). However, in the UFC Daniel Cormier had won 45% of his fights via KO/TKO (Sherdog.com, n.d.). Cormier also possesses an above average significant strikes landed per minute score of 4.25. Although renowned for his grappling, Daniel Cormier has career statistics more suited for striking.

Dan Henderson is known as an elite striker with 53% of his victories coming by way of KO/TKO (Sherdog.com, n.d.). However, within the SOM's he was frequently clustered alongside grapplers. The reason for this is even though Henderson wins most of his fights through strikes, he has a below average number of significant strikes landed per minute of 2.44. These values suggest that although Henderson has a low output of strikes, he is selective and clinical with them. Dan Henderson does also have grappling credentials being a former Olympic wrestler and Pan American Games gold medalist (Abbott, 2016).

The outcome of the results showed the 16x16 SOM trained at 1,000 iterations could correctly differentiate between strikers and grapplers. With a silhouette score of 0.577, the values in the SOM were clustered together and resemble strongly the values surrounding them.

Once evaluated, it was clear that the characteristics of the outliers in the SOM did display features of the opposing fighting style. Henderson possesses career statistics comparable to grapplers, whereas Cormier's statistics resembled that of a striker. The SOM can categorise these fighters accurately due to the characteristics of fighters.

Fighters who were considered strikers typically would possess above average significant strike output, accuracy, defense and takedown defense. The above average takedown defense indicates strikers were well equipped and prepared to prevent the fight going to the ground. Fighters considered grapplers have above average significant strike defense, takedown average, takedown defense and submission averages. Grapplers have below average significant strike output, absorption and takedown accuracy **(Table 2)**.

Table 2. Career statistic averages for the seven selected strikers and grapplers, along with a cumulative average of all fighters in the fighter dataset.

Fighter Type	SLpM	Str. Acc	SApM	Str. Def	TD Avg	TD Acc	TD Def	Sub. Avg
Striker	3.83	49.29	2.84	58.29	0.85	50.29	70.86	0.47
Average Fighter	3.04	43.19	3.09	55.36	1.60	39.76	57.83	0.74
Grappler	2.98	47.57	2.40	58.43	2.55	36.86	59.71	1.31

4.3 Limitations

When investigating competitive sport related topics, it's impossible to ignore the factor of unpredictability, there's no guarantee on how the outcome of an event will play out due to environmental and individual variables (Torrents *et al.*, 2017). Injuries, age and mental states can mean favorites in a bout can lose or persuade them to fight in an unorthodox way (Appleton and Hill, 2012). These instances of out of the ordinary performances can confuse the career statistics of a fighter, meaning categorising the fighter correctly can become impossible with solely the values alone.

Whenever a bout is started, the fighters always start on their feet, hence grapplers were immediately at a disadvantage. If a grappler is unable to take the opponent down then the grappler will have no other choice but to exchange in a striking battle, which can potentially change the classification of a fighter. If a grappler has several UFC fights where they have been

unable to take their opponent down, regardless of their prior credentials, it is unlikely a SOM would classify them as a grappler.

Professional MMA fighters may compete in other organisations before entering the UFC, other large MMA organisations (Bellator, One, PFL and Rizin) do not store fighter statistics of their fighters, any past success or style a fighter has created prior to entering the UFC will not be considered, meaning any poor performances in the UFC won't be cushioned by the fighters previous career statistics.

4.4 Future Use

Implementation of SOM architecture in the future for fighter classification could investigate into classification of fighting disciplines (boxing, wrestling and Muay Thai). More refined input values would be necessary to classify the disciplines. The UFC statistics page stores detailed analytics for every bout in the organisation. Unlike the fighter's pages, the bouts differentiate between significant strikes the head, body, kicks and ground strikes. If each fighter could have the detailed analytics attached to their fighter profile, a more detailed overview of the fighter's patterns can be studied and with this data it could be possible to not only distinguish between strikers and grapplers, but also boxers and kickboxers.

The use of this architecture could benefit MMA fighters, when training for a bout understanding the opponents' patterns and techniques gives the fighter a significant advantage (Baker and Schorer, 2013).

SOMs may potentially evaluate an opponent and give a breakdown of the fighter's analytics and show fighters they were most like, giving more ability to prepare for the opponent's style.

4.5 Conclusion

The investigation conducted showed that a SOM with statistics of a fighter can accurately classify a striking based fighter or a grappling based fighter. Fighters can be widely considered to be of a specific style. However, when their statistics were compared, the fighter's results can resemble that of a different fighting style. The main limitations of the investigation were the unpredictability of competitive sports, no matter how much a team or individual is deemed the favorite, there will always be uncertainty in sport due to a multitude of factors and human error. Through testing, the investigation found the optimal solution for neighborhood size and iteration length for classifying fighters, a 16x16 node map iterated 1,000 times provides neighborhoods where the values in them were similar in attributes and behavior. Machine learning is becoming more prominent within sport and SOM architecture can benefit competitors by understand the opponent more, increasing the chances of victory (Ding, 2019).

With technological and physical breakthroughs, MMA has a larger talent pool than ever, the gap between elite fighters grows continuously smaller. Machine learning has the potential to change they way fighters train for an opponent and transform the way fighters are classified in MMA. Instead of seeing fighters as just boxers or wrestlers, SOM architecture can deeply analyse and classify the type of martial artist a fighter is when fighting in MMA bouts.

7. Reference List

Abbott, G., 2016. *Wrestlers In MMA: Olympian Dan Henderson Trains For His Final Fight Vs. Michael Bisping In UFC 204 Main Event*. [online] Team USA. Available at: <https://www.teamusa.org/USA-Wrestling/Features/2016/September/23/Dan-Henderson-to-finish-MMA-career-in-title-fight-vs-Bisping> [Accessed 18 April 2020].

Alpaydin, E., 2020. *Introduction To Machine Learning*. 4th ed. Ethem Alpaydin.

ALONSO, M., 2013. VALE TUDO: A RICH, STORIED & COMPLEX PAST.

Appleton, P. and Hill, A., 2012. Perfectionism and Athlete Burnout in Junior Elite Athletes: The Mediating Role of Motivation Regulations. *Journal of Clinical Sport Psychology*, 6(2), pp.129-145.

Baker, J. and Schorer, J., 2013. The Southpaw Advantage? - Lateral Preference in Mixed Martial Arts. *PLoS ONE*, 8(11), p.e79793.

Bet-bonuscode.co.uk. 2020. How AI Is Changing The Betting Industry. [online] Available at: <https://www.bet-bonuscode.co.uk/how-is-ai-changing-betting-industry/> [Accessed 8 April 2020].

Brownlee, J., 2020. *Supervised And Unsupervised Machine Learning Algorithms*. [online] Machine Learning Mastery. Available at: <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/> [Accessed 9 April 2020].

Chu, C., 1989. Cluster analysis in manufacturing cellular formation. *Omega*, 17(3), pp.289-295.

Chu, W., Shih, C., Chou, W., Ahamed, S. and Hsiung, P., 2019. Artificial Intelligence of Things in Sports Science: Weight Training as an Example. Computer, 52(11), pp.52-61.

Crummy.com. 2020. *Beautiful Soup Documentation — Beautiful Soup 4.9.0 Documentation.* [online] Available at: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/> [Accessed 8 April 2020].

Ding, P., 2019. Analysis of Artificial Intelligence (AI) Application in Sports. *Journal of Physics: Conference Series*, 1302, p.032044.

Encyclopedia Britannica. 2020. *Mixed Martial Arts*. [online] Available at: <https://www.britannica.com/sports/mixed-martial-arts> [Accessed 7 April 2020].

Gamblingcommission.gov.uk. 2020. Industry Statistics. [online] Available at: <https://www.gamblingcommission.gov.uk/news-action-and-statistics/Statistics-and-research/Statistics/Industry-statistics.aspx> [Accessed 8 April 2020].

Gan, G. and Ng, M., 2017. k -means clustering with outlier removal. *Pattern Recognition Letters*, 90, pp.8-14.

Gupta, J., 2019. The Role of Artificial intelligence in Agriculture Sector. *customer think*,.

Kearn, M., 2016. *Machine Learning Is For Muggles Too*. [online] Martink.me. Available at: <http://martink.me/articles/machine-learning-is-for-muggles-too> [Accessed 8 April 2020].

Grunz, A., Memmert, D. and Perl, J., 2012. Tactical pattern recognition in soccer games by means of special self-organizing maps. *Human Movement Science*, 31(2), pp.334-343.

Hirose, A. and Pih, K., 2009. Men Who Strike and Men Who Submit: Hegemonic and Marginalized Masculinities in Mixed Martial Arts. *Men and Masculinities*, 13(2), pp.190-209.

James, L., Haff, G., Kelly, V. and Beckman, E., 2016. Towards a Determination of the Physiological Characteristics Distinguishing Successful Mixed Martial Arts Athletes: A Systematic Review of Combat Sport Literature. *Sports Medicine*, 46(10), pp.1525-1551.

Kohonen, T., 2013. Essentials of the self-organizing map. *Neural Networks*, 37, pp.52-65.

Kulkarni, P., Londhe, S. and Deo, M., 2017. Artificial Neural Networks for Construction Management: A Review. Soft Computing in Civil Engineering, 1(2).

Lamb, P. and Croft, H., 2016. Visualizing Rugby Game Styles Using Self-Organizing Maps. IEEE Computer Graphics and Applications, 36(6), pp.11-15.

Lystad, R., Gregory, K. and Wilson, J., 2014. The Epidemiology of Injuries in Mixed Martial Arts. Orthopaedic Journal of Sports Medicine, 2(1), p.232596711351849.

Meyer-Baese, A. and Schmid, V., 2014. *Pattern Recognition And Signal Analysis In Medical Imaging*. 2nd ed. Academic Pr.

Majumder, A., Behera, L. and Subramanian, V., 2014. Emotion recognition from geometric facial features using self-organizing map. *Pattern Recognition*, 47(3), pp.1282-1293.

Olszewski, D., 2014. Fraud detection using self-organizing map visualizing the user profiles. *Knowledge-Based Systems*, 70, pp.324-334.

Patil, A., 2020. *Artificial Intelligence (AI) Market Outlook: 2025*. [online] Allied Market Research. Available at: <https://www.alliedmarketresearch.com/artificial-intelligence-market> [Accessed 10 April 2020].

Ray, P., 2016. A Survey on Internet of Things Architectures. *EAI Endorsed Transactions on Internet of Things*, 2(5), p.151714.

Ratamess, N., 2011. Strength and Conditioning for Grappling Sports. *Strength and Conditioning Journal*, 33(6), pp.18-24.

Sesker, C., 2014. *Past World Medalist Daniel Cormier Overcomes Adversity To Earn UFC Title Shot*. [online] Team USA. Available at: <https://www.teamusa.org/USA-Wrestling/Features/2014/September/04/Past-World-medalist-Daniel-Cormier-overcomes-adversity-to-earn-UFC-title-shot> [Accessed 11 April 2020].

Sherdog. n.d. *Sherdog.Com*. [online] Available at: <https://www.sherdog.com/fighter/Daniel-Cormier-52311> [Accessed 11 April 2020].

Spanias, C., Nikolaidis, P., Rosemann, T. and Knechtle, B., 2019. Anthropometric and Physiological Profile of Mixed Martial Art Athletes: A Brief Review. *Sports*, 7(6), p.146.

Springenberg, J., 2016. UNSUPERVISED AND SEMI-SUPERVISED LEARNING WITH CATEGORICAL GENERATIVE ADVERSARIAL NETWORKS. *Cornell Univserity*.

Sun, L., Chen, G., Xiong, H. and Guo, C., 2017. Cluster Analysis in Data-Driven Management and Decisions. *Journal of Management Science and Engineering*, 2(4), pp.227-251.

Statista. 2020. *Average PPV Buys Per Event 2001-2018 | Statista*. [online] Available at: <https://www.statista.com/statistics/861468/ultimate-fighting-championship-average-ppv-buys-per-event/> [Accessed 10 April 2020].

Torrents, C., Passos, P. and Cos, F., 2017. Complex Systems in Sport, International Congress: Linking Theory and Practice. *Complex Systems in Sport, International Congress: Linking Theory and Practice*, 5.

Ufc.com. 2020. Unified Rules Of Mixed Martial Arts. [online] Available at: <https://www.ufc.com/unified-rules-mixed-martial-arts> [Accessed 13 April 2020].

Ultsch, A. and Morchen, F., 2005. ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM.

Vargiu, E. and Urru, M., 2012. Exploiting web scraping in a collaborative filtering- based approach to web advertising. *Artificial Intelligence Research*, 2(1).

Vettigli, G., 2018. *MiniSom.* [online] GitHub. Available at: <https://github.com/JustGlowing/MiniSom> [Accessed 4 April 2020].

Zheng, C., He, G. and Peng, Z., 2015. A Study of Web Information Extraction Technology Based on Beautiful Soup. *Journal of Computers*, 10(6), pp.381-387.

8. Appendices

Appendices 1. A heatmap for every neighborhood and iteration test done within the investigation.